

## PETSHOP ONLINE (vanzari hrana)

### **Componente:**

Aplicatia are doua componente:

-partea de administrare in care un admin poate insera, modifica pretul sau chiar sterge produse. In aceasta parte nu se pot face cumparaturi. Aici exista un singur Superuser care poate sterge un utilizator, poate vizualiza toti userii existenti, poate adauga un admin si poate vizualiza diverse rapoarte (interogarile simple si complexe). Acest Superuser poate de asemenea sa insereze, sa modifice pretul si sa stearga produse.

-partea destinata utilizarii de care un user normal in care se pot crea conturi si se pot face cumparaturi.

### **Functionalitati:**

Pentru admin:

- Butoane pentru inserarea/stergerea sau modificarea pretului unui produs
- Posibilitate de schimbare a parolei contului
- Logout

Pentru Superuser:

- Butoane pentru inserarea/stergerea sau modificarea pretului unui produs
- Posibilitate de schimbare a parolei contului
- Logout
- Adaugare de admini
- Stergere de useri sau de admini
- Vizualizarea tuturor userilor
- Informatii aditionale (interogarile)

Pentru user normal:

- Crearea unui cont
- Posibilitate de schimbare a parolei contului
- Logout
- Butoane pentru adaugarea in cos a produselor dorite
- Vizualizarea cosului de cumparaturi + plasare comanda

### Baza de date:

- Tabele:

Superuser: contine datele personale ale utilizatorilor, username-ul si parola, precum si drepturile de acces (UserAddRights = 'Y' => superuser, Admin = 'Y' => poate insera/sterge si modifica produse. Daca amandoua sunt 'N' atunci userul poate face doar cumparaturi de pe site).

Produse: contine denumirea produsului, pret, cantitate, imaginea, codul produsului, numarul de produse pe stoc din acel tip. Campul promotie este Y cand se doreste display-ul pe pagina principala a produsului (setarea produsului ca fiind promotional), altfel este N.

Campurile ID\_categorie si ID\_hrana sunt chei straine din tabele Categorie si Hrana.

Categorie: contine tipul animalului pentru fiecare produs (caine,pisica,etc).

Hrana: contine tipul de hrana pentru fiecare produs (uscata,umeda,etc).

Legatura: rezolva problema produselor pe un singur bon impreuna cu tabela Cumparaturi.

Chei straine: ID\_prod: ID-ul produsului cumparat si ID\_cump:ID-ul bonului pe care este inserat.

Cumparaturi: contine totalul de plata si ID-ul userului care a plasat comanda.

Chei straine: ID\_user

- Campuri si tipuri de date:

Superuser:

	Column Name	Data Type	Allow Nulls
🔑	ID_sup	int	<input type="checkbox"/>
	Nume	nvarchar(50)	<input type="checkbox"/>
	Prenume	nvarchar(50)	<input type="checkbox"/>
	CNP	char(13)	<input type="checkbox"/>
	Email	nvarchar(50)	<input type="checkbox"/>
	Username	nvarchar(50)	<input type="checkbox"/>
	Parola	nvarchar(50)	<input type="checkbox"/>
	UserAddRights	char(1)	<input type="checkbox"/>
	Admin	char(1)	<input type="checkbox"/>
			<input type="checkbox"/>

Produce:

	Column Name	Data Type	Allow Nulls
▶	ID_prod	int	<input type="checkbox"/>
	Denumire	nvarchar(60)	<input type="checkbox"/>
	ID_categorie	int	<input type="checkbox"/>
	[Pret/kg]	numeric(7, 3)	<input type="checkbox"/>
	ID_hrana	int	<input type="checkbox"/>
	Cantitate	numeric(7, 3)	<input type="checkbox"/>
	Img	varchar(50)	<input type="checkbox"/>
	Cod_produs	varchar(10)	<input type="checkbox"/>
	Promotie	char(2)	<input type="checkbox"/>
	Stoc	int	<input type="checkbox"/>
			<input type="checkbox"/>

Categorie:

	Column Name	Data Type	Allow Nulls
▶	ID_Categorie	int	<input type="checkbox"/>
	Tip_animal	nvarchar(50)	<input type="checkbox"/>
			<input type="checkbox"/>

Hrana:

	Column Name	Data Type	Allow Nulls
▶	ID_hrana	int	<input type="checkbox"/>
	Tip_hrana	nvarchar(50)	<input type="checkbox"/>
			<input type="checkbox"/>

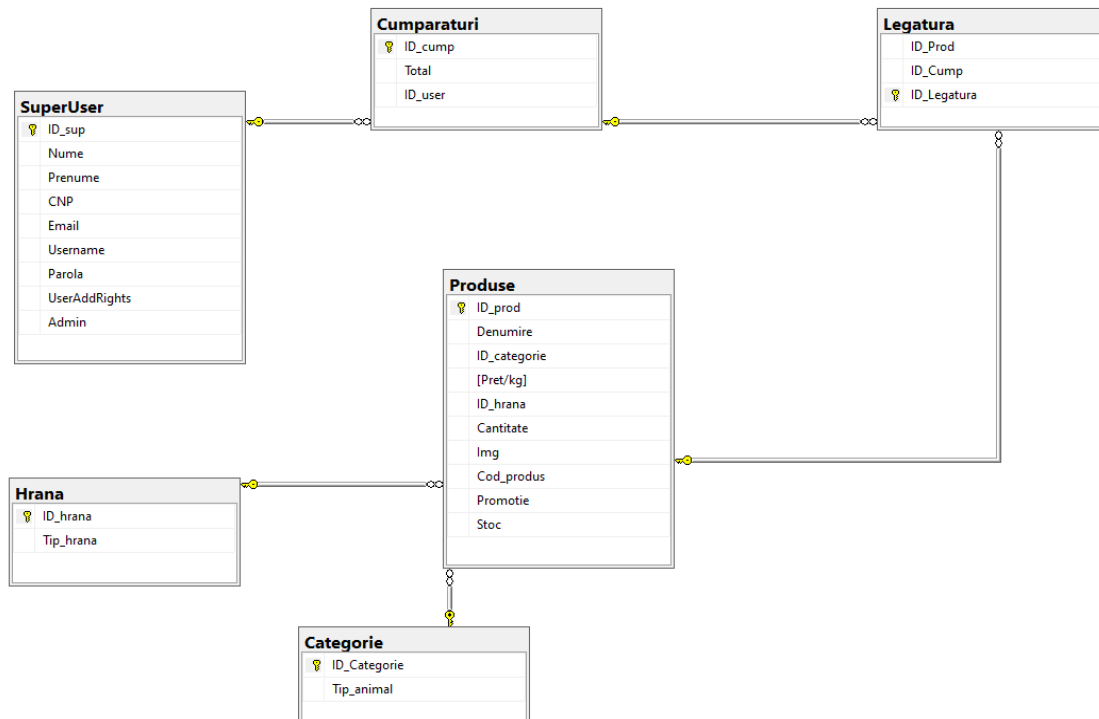
Legatura:

	Column Name	Data Type	Allow Nulls
▶	ID_Prod	int	<input type="checkbox"/>
	ID_Cump	int	<input type="checkbox"/>
▶	ID_Legatura	int	<input type="checkbox"/>
			<input type="checkbox"/>

## Cumparaturi:

Column Name	Data Type	Allow Nulls
<b>ID_cump</b>	int	<input type="checkbox"/>
Total	numeric(7, 2)	<input type="checkbox"/>
ID_user	int	<input type="checkbox"/>
		<input type="checkbox"/>

- Relatii intre tabele:



- Constrangeri:

Pe langa constrangerile de cheie straina prezentate in sectiunea Tabele, am impus si o constrangere delete cascade:

```
alter table Legatura
add constraint fk_cumparaturi
Foreign key (ID_cump)
references Cumparaturi(ID_cump)
on delete cascade
```

La stergerea unui bon se sterg si produsele de pe el. Proprietatea aceasta este folosita atunci cand un user iese din program (logout) fara sa finalizeze cumparaturile.

### **Interfata vizuala:**

Am ales sa realizez interfata in Visual Studio 2019 , C#. Pentru documentatie cod sursa vezi *Readme*.

### **Insert:**

- Adaugarea unui utilizator(prin register) sau a unui admin (de catre superuser)

```
Insert Into SuperUser values('n','p','n','em','u','ps','Y','N')
```

n = nume; p = prenume; c = CNP; em = email; u = username; ps = parola; add user rights si admin rights

- Initializarea totalului de plata cu 0 la inceperea cumparaturilor:

```
Insert into Cumparaturi values('0','id')
```

id = id-ul userului current. Este preluat din tabela Superuser dupa etapa de login.

- Adaugarea unui produs pe bon:

```
Insert into Legatura values('id_prod','id_bon')
```

- Adaugarea unui produs in tabela Produse (de catre admin sau superuser):

```
Insert Into Produse values(' den ',' categorie','pret ',' hrana',' cant',' nume_img', 'cod' , ' prom
','numericUpDown1.Value')
```

numericUpDown value reprezinta nr de produse introduse de admin pe stoc.

### Update:

- Schimbarea parolei:

Update SuperUser set Parola ='p2' where Username ='user'

p2 = noua parola aleasa de utilizator

user = usernameul userului care este logat pe site

- Scaderea nr. De produse aflate pe stoc la adaugarea unui in cos:

Update Produse Set Stoc = (stoc - 1) where Denumire=' den'

stoc = nr de produse de un anumit tip pe stoc. In aceasta variabila se stocheaza initial valoarea din tabela Produse->stoc

den = numele produsului

- Actualizarea totalului de plata pe masura ce se adauga produse in cos:

Update Cumparaturi set Total = "total" where ID\_cump = "id\_bon"

- Modificarea pretului unui produs(de catre admin sau superuser):

Update Produse Set [Pret/kg] = ' p ' where Denumire like '% textBox1.Text.ToString()%' or Cod\_produs like '% textBox1.Text.ToString() %'";

Adminul introduce denumirea sau codul produsului care urmeaza sa fie sters in textboxul1.  
Denumirea/codul este preluat si se face o cautare in bd pentru a verifica daca produsul introdus exista, dupa care este modificat pretul.

- Actualizarea totalului la stergerea unui produs din cos:

Update Cumparaturi set Total=" total" where ID\_cump=" id\_bon"

Inainte de update din variabila total se scade pretul produsului care tocmai a fost sters din cos.

### Delete:

- Stergerea unui utilizator (de catre superuser):

Delete From SuperUser where Username like 'username'

- Stergerea unui produs(de catre admin):

Delete from Produse where Denumire like '%den%' or Cod\_produs = 'den'

den = inputul dat de admin (denumirea sau codul produsului care urmeaza sa fie sters)

- Stergerea unui produs din cos:

SET ROWCOUNT 1 Delete from Legatura where ID\_Prod =" id\_prod" and ID\_cump=" id\_bon "

SET ROWCOUNT 0

Am folosit set rowcount pentru a sterge doar un produs din cos atunci cand sunt mai multe de acelasi fel.

### **Interogari simple:**

- Produsele care au pretul/kg > 50 ordonate descrescator dupa pret

```
Select P.Denumire, P.[Pret/kg]
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
join Hrana H on P.ID_hrana=H.ID_hrana
where P.[Pret/kg]>50 order by P.[Pret/kg] DESC
```

- Total stoc pt fiecare categorie de animal

```
Select Cat.Tip_animal,Sum(P.Stoc) as Total_stoc
from Produse P join Categorie Cat on Cat.ID_Categorie=P.ID_categorie
group by Cat.Tip_animal
order by Sum(P.Stoc)
```

- Cate produse s-au vandut din fiecare tip

```
Select count(L.ID_Cump) as
Nr_produce,Denumire,(P.[Pret/kg])*P.Cantitate as Pret_produs,
Cat.Tip_animal,H.Tip_hrana
  from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
                        join Hrana H on P.ID_hrana=H.ID_hrana
                        join Legatura L on P.ID_prod=L.ID_Prod
group by
P.Denumire,(P.[Pret/kg])*P.Cantitate,Cat.Tip_animal,H.Tip_hrana
order by count(L.ID_Cump)
```

- Cate produse aflate la promotie sunt pe fiecare bon

```
Select C.ID_cump,count(L.ID_Cump) as Nr_produce
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
                        join Hrana H on P.ID_hrana=H.ID_hrana
                        join Legatura L on L.ID_prod=P.ID_Prod
                        join Cumparaturi C on C.ID_Cump = L.ID_cump
group by C.ID_cump,P.Promotie
having P.Promotie='Y'
order by C.ID_Cump
```

- Afisarea tuturor produselor dintr-o anumita categorie

```
select P.Denumire,(P.[Pret/kg])*(P.Cantitate) as Pret,P.Cantitate*1000
as Greutate, P.Img, P.Cod_Produs
  from Produse P join Categorie C on P.ID_categorie = C.ID_categorie
                        join Hrana H on P.ID_hrana = H.ID_hrana
                        where C.Tip_animal = 'Caine' and H.Tip_hrana
                        = 'Uscata'
```

Aceasta interogare se repeta pentru fiecare tip de animal si de hrana.

- Afisarea tuturor produselor aflate in cosul de cumparaturi

```
select P.Denumire,(P.[Pret/kg])*Cantitate as Pret,P.Cantitate*1000 as
Cantitate, L.ID_Prod , C.ID_cump,L.ID_Prod
from Legatura L join Cumparaturi C on L.ID_cump = C.ID_cump
                        join Produse P on L.ID_Prod = P.ID_prod
where C.ID_cump = " Form1.id_bon";
```



### Interogari complexe:

- Toate produsele care se afla pe bonurile cu totalul mai mare de 100 de lei

```
Select ID_Cump,Denumire,Cod_produs,Cat.Tip_animal
from Produse P join Legatura L on L.ID_Prod=P.ID_prod
      join Categorie Cat on Cat.ID_Categorie=P.ID_categorie
where L.ID_Cump in(Select ID_Cump
                  from Cumparaturi C
                  where C.Total>100)
```

- Toate produsele care se afla pe bonurile care au mai putin sau egal 3 produse

```
Select L.ID_cump, P.Denumire,(P.[Pret/kg])*P.Cantitate as Pret_produs,
Cat.Tip_animal,H.Tip_hrana
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
      join Hrana H on P.ID_hrana=H.ID_hrana
      join Legatura L on P.ID_prod=L.ID_Prod
where L.ID_Cump in (select C.ID_cump from Cumparaturi C join Legatura L
on L.ID_Cump=C.ID_cump
                    group by C.ID_cump having
count(L.ID_Cump)<=3)
```

- Produsele care nu se afla pe nici un bon

```
Select P.Denumire,(P.[Pret/kg])*P.Cantitate as Pret_produs,
Cat.Tip_animal,H.Tip_hrana
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
      join Hrana H on P.ID_hrana=H.ID_hrana

where P.ID_prod not in (select distinct L.ID_prod from Cumparaturi C
join Legatura L on L.ID_Cump=C.ID_cump)
```

- Toate produsele cumparate de userul 'test1'

```
Select P.Denumire,(P.[Pret/kg])*P.Cantitate as Pret_produs,
Cat.Tip_animal,H.Tip_hrana
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
      join Hrana H on P.ID_hrana=H.ID_hrana
      join Legatura L on P.ID_prod = L.ID_Prod
      join Cumparaturi C on L.ID_Cump=C.ID_cump
      join SuperUser U on U.ID_sup=C.ID_user
where C.ID_user in (Select ID_sup from Superuser where
Username='test1' )
```

- Produsele din categoria hrana uscata caine(variabil) care se afla pe bonurile cu totalul >120 (variabil)

```
Select ID_Cump,Denumire,Cod_produs,Cat.Tip_animal,H.Tip_hrana
from Produse P join Legatura L on L.ID_Prod=P.ID_prod
                        join Categorie Cat on
Cat.ID_Categorie=P.ID_categorie
                        join Hrana H on P.ID_hrana=H.ID_hrana
where L.ID_Cump in(Select ID_Cump
                    from Cumparaturi C
                    where C.Total>50) and Cat.Tip_animal='Pasari' and
H.Tip_hrana='Hrana'
```

- Toate produsele cumparate de userul 'test2' care se afla pe bonurile care au mai mult de 2 produse

```
Select L.ID_cump, P.Denumire,(P.[Pret/kg])*P.Cantitate as Pret_produs,
Cat.Tip_animal,H.Tip_hrana
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
                        join Hrana H on P.ID_hrana=H.ID_hrana
                        join Legatura L on P.ID_prod=L.ID_Prod
                        join Cumparaturi C on L.ID_Cump= C.ID_cump
                        join SuperUser U on C.ID_user=U.ID_sup
where L.ID_Cump in (select C.ID_cump from Cumparaturi C join Legatura L
on L.ID_Cump=C.ID_cump
                        group by C.ID_cump having
count(L.ID_Cump)>2) and U.ID_sup in (select ID_sup from Superuser where
Username='test2')
```

- Toate produsele care se afla pe bonul cu totalul cel mai mare

```
Select P.Denumire,(P.[Pret/kg])*P.Cantitate as Pret_produs,
Cat.Tip_animal,H.Tip_hrana,C.Total
from Produse P join Categorie Cat on P.ID_categorie=Cat.ID_Categorie
                        join Hrana H on P.ID_hrana=H.ID_hrana
                        join Legatura L on P.ID_prod=L.ID_Prod
                        join Cumparaturi C on L.ID_Cump = C.ID_cump
                        where C.Total IN (Select max(Total) from Cumparaturi)
```